AD-A035 209 HARRY DIAMOND LABS ADELPHI MD F/G 9/2
USERS MANUAL FOR THE MODULAR ANALYSIS-PACKAGE LIBRARIES ANAPAC --ETC(U)
NOV 76 T V NOON
UNCLASSIFIED HDL-TR-1782 NL

1 OF 1
ADA035209

END
DATE
FILMED
3-77

(11)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER (14) HDL-TR-1782	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER (9)
4. TITLE (and Subtitle) (6) Users' Manual for the Modular Analysis-Package Libraries ANAPAC and TRANL		5. TYPE OF REPORT & PERIOD COVERED Technical Report
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) (10) Thomas V. Noon		8. CONTRACT OR GRANT NUMBER(s) (16) DA: 1W162118AH75/C
9. PERFORMING ORGANIZATION NAME AND ADDRESS Harry Diamond Laboratories 2800 Powder Mill Road Adelphi, MD 20783		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Program: 6.21.18.A
11. CONTROLLING OFFICE NAME AND ADDRESS US Army Materiel Development & Readiness Command Alexandria, VA 22333		12. REPORT DATE (11) November 1976
		13. NUMBER OF PAGES 49 (12) 45p.
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

HDL Project: X756E2 DRCMS Code: 612118l1H7500
This work was funded under DARCOM NWER/T project, titled Multiple Systems Evaluation Program.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Computer software
Computer software library
Digitized data
Plotting software
Fourier transform
Interpolation software

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A user's manual for two computer software libraries is presented. The computer libraries, ANAPAC and TRANL, are a collection of modular computer programs and subroutines. The libraries contain computer routines for plotting, for interpolation, for calculating Fourier transforms, for processing digitized data, and for structuring data files. The purpose and implementation of the programs and subroutines are presented. over

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED 163 050

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

bpg

In addition, the application of these modular programs to the Generic Assessment Methods for a Priori Hardening Systems (GAMPHS) is explained. The GAMPHS approach provides proven analytical computer programs to commodity commands or project managers for EMP assessments of Army systems.

CONTENTS

CONTENTS (CONT'D)

FIGURES

1. INTRODUCTION

Nuclear weapons deployed at high altitudes produce high-altitude electromagnetic pulse effects (HEMP) that can seriously degrade tactical weapon and communication systems vitally needed by the field army prepared to fight a conventional and nuclear war. The Multiple Systems Evaluation Program (MSEP) was established to determine both the vulnerability and the means for hardening a large number of U.S. Army tactical systems to an HEMP environment. An essential step in the program is to develop analytic tools to evaluate system susceptibility to HEMP, that is, computer programs for predicting transient data and system response. These computer programs have been gathered into an applications package titled GAMPHS[1] (Generic Assessment Methods for a Priori Hardening of Systems). The GAMPHS application for the vulnerability and hardness assessment of systems covered by MSEP uses two modular computer libraries, TRANL and ANAPAC, in addition to other computer programs (fig. 1). These library routines are the part of GAMPHS, whose purpose is to provide proven analytical computer programs to commodity commands or project managers for assessment of the EMP vulnerability of Army systems. The application of TRANL and ANAPAC to GAMPHS is explained in section 4 of this report.

The TRANL and ANAPAC modular analysis package libraries are a collection of independent computer software programs and subroutines, each written to do a specific task, and stored in two permanent files that form user's libraries. The modular building blocks in ANAPAC can be combined in the user's computer program to meet his needs. Most of the computational building blocks are input/output-free to allow the user greater flexibility in writing a driver program. Two of the modular subroutines, RDTAPE and WRTAPE, are provided for input/output compatibility of data files written on either disc or tape. Except for the fast Fourier transform algorithm, the software is written in FORTRAN IV. The fast Fourier transform algorithm is written in COMPASS assembly language, which noticeably decreases its execution time.

The TRANL library is a collection of independent programs that primarily supports the data digitizing system. The programs can be merged into the control streams of major programs to preprocess digitized data and prepare the data for analysis. The data structure of the TRANL programs is compatible with RDTAPE and WRTAPE. The use of programs TEDD and REDUCE is not restricted to digitized data. All the programs in TRANL are written in FORTRAN IV.

---

[1]*George Gornak et al, EMP Assessment for Army Tactical Communications Systems: Transmission Systems Series No. 1--Radio Terminal Set AN/TRC-145 (U), Harry Diamond Laboratories TR-1746 (February 1976). (SECRET-RESTRICTED DATA)*

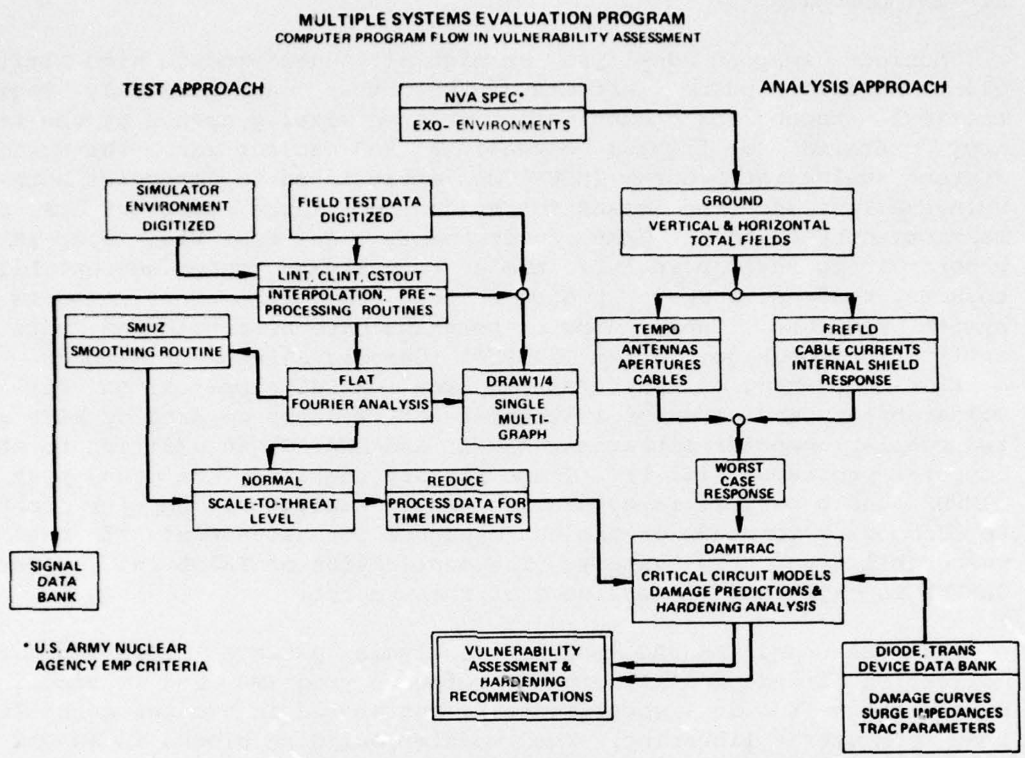

Figure 1. The GAMPHS application in vulnerability and hardness.

The ANAPAC and TRANL libraries are stored on the Control Data Corporation (CDC) Computer System at the U.S. Army Mobility Equipment Research and Development Command (USAMERADCOM) and accessed by means of the ATTACH and LIBRARY Scope control cards. The subroutines in ANAPAC are accessed by a standard FORTRAN CALL statement. The programs in TRANL can be accessed only through the control cards.

2. DESCRIPTION OF PROGRAMS AVAILABLE IN THE ANAPAC LIBRARY

2.1 Plotting Routines

The plotting software presented here was written for use with a Houston Instrument DP-1 plotter on line to a Mohawk 2400 system. The Mohawk system is used as a remote job entry (RJE) station to a CDC 6600.

2.1.1 Subroutine DRAW1

The FORTRAN callable computer subprogram DRAW1[2] is designed to plot a single graph of one function (data set) complete with axes, necessary scaling, and user-defined annotations (axis labels, plot title, and plot subtitle). The plot can be a linear-linear plot, log-log plot, linear-log plot, or log-linear plot. When arrays are to be plotted on a log axis, the software will drop values that are invalid for a logarithmic scale (that is, zero or negative values) when they occur at the beginning of an array; a message is then printed that notifies the user what has been done. Invalid values occurring within an array cause the termination of the plot, and an appropriate message is printed. The date and time of execution for the processing of each graph are displayed in the lower right-hand corner of the graph. The first time DRAW1 is called, a remote operator message (in this system, a "PM message") is generated that alerts the operator that THIS JOB PLOTS.

User-defined annotation refers to axis labels, plot title, and plot subtitle, all of which are optional. Axis labels may contain up to three words (30 characters). When an array contains axis labels that are greater than three words, the output to the plotter is unpredictable. Besides the user-defined axis notations, the plotting software adds a scaling factor after any label, if necessary, to prevent the numerical notations on the axis from overlapping illegibly. This scaling factor notation is recorded in the following format:

$$\text{"(X } 10^{\pm N}\text{)"}$$

The plot title and subtitle may each consist of up to eight words (80 characters), placed at the top of the plot. The subtitle is placed below the title in slightly smaller print. To center the title or subtitle, the user fills the appropriate arrays with leading blanks. The user can incorporate special characters (see fig. 2) into the title and subtitle by punching the corresponding Hollerith punch in the appropriate character position in the title and subtitle arrays.

For the standard and default axis position, the plotter places the two axes so that they intersect in the lower left corner of the graph. For arrays having both positive and negative values, two other options are available for the linear axis. The axis can be drawn either at the zero point occurrence or at the standard location with a parallel line at the zero occurrence as a reference line. If these options are selected and the array values do not span zero, the standard axis position is selected by default.

---

[2]*Thomas V. Noon, Enhanced Plotting Software for Use with the Houston Instrument Complot Plotter, Harry Diamond Laboratories TM-75-32 (December 1975).*

SYMBOL	026 PUNCH	029 PUNCH
#	0-8-6	8-3
[	12-8-2	12-8-2
!	11-8-2	11-8-2
%	8-6	0-8-4
"	8-4	8-7
△	0-8-5	0-8-5
]	0-8-2	12-8-7
&	0-8-7	12
'	8-7	8-5
?	11-8-6	0-8-7
<	8-5	12-8-4
>	11-8-7	0-8-6
@	11-8-5	8-4
□	12-8-5	0-8-2
⧖	12-8-6	11-8-7
;	12-8-7	11-8-6
:	8-2	8-2

Figure 2. Hollerith code for special characters.

If the number of points to be plotted is passed to DRAW1 as a positive value, the data set arrays are restored to the values they had before being passed to DRAW1. This is done by recalculating the values. If the values of the data set arrays are no longer needed, the number of points to be plotted can be passed as a negative value. This will result in the data arrays *not being restored*, thus saving some execution time.

The FORTRAN subprogram DRAW1 is called as follows:

```
CALL DRAW1(ITYPE,IXLAB,IYLAB,IPTLAB,ISBLAB,NPT,
           ZAX,ZAY,XLAB,YLAB,PTLAB,SUBLAB,X,Y)
```

The first six formal parameters in the subroutine argument list are integer variables; the remaining eight are real variables. The definition and use of each is as follows:

ITYPE Variable used to select the type of plot.

If ITYPE=1 a linear-linear plot is selected.

=2 a log-log plot is selected.

=3 a linear-log plot (semilog with the linear axis for the dependent array) is selected.

=4 a log-linear plot (semilog with the linear axis for the independent array) is selected.

IXLAB Number of words in the label for the independent axis, not to exceed 3. If IXLAB=0, axis label is omitted.

IYLAB Number of words in the label for the dependent axis, not to exceed 3. If IYLAB=0, axis label is omitted.

IPTLAB Number of words in the plot title, not to exceed 8. If IPTLAB=0, plot title is omitted.

ISBLAB Number of words in the plot subtitle, not to exceed 8. If ISBLAB=0, plot subtitle is omitted.

NPT Number of points to be plotted in both the independent and dependent data arrays. If the negative of the number of points is passed through NPT, the data arrays are not restored to the values they had before being passed.

ZAX Variable that determines where the independent axis is drawn.

If ZAX=0. axis is drawn in the standard location. This is also the default position when the next two conditions cannot be met.

=1, axis is drawn at the zero occurrence of the dependent axis.

=2, axis is drawn at a standard location with a zero reference line placed at the zero occurrence on the dependent axis.

ZAY Variable that determines where the dependent axis is drawn.

If ZAY=0, axis is drawn in standard location. This is also the default position when the next conditions cannot be met.

=1. axis is drawn at the zero occurrence of independent axis.

=2. axis is drawn at standard location with the addition of a zero reference line placed at the zero occurrence on the independent axis.

XLAB Array containing user-defined label for the independent axis, not to exceed 3 words (30 characters). If IXLAB=0, a dummy name can be used in the call.

YLAB Array containing user-defined label for the dependent axis, not to exceed 3 words (30 characters). If IYLAB=0, a dummy name can be used in the call.

PTLAB Array containing user-defined title for the plot, not to exceed 8 words (80 characters). If IPTLAB=0, a dummy name can be used in the call.

SUBLAB Array containing user-defined subtitle for the plot, not to exceed 8 words (80 characters). If ISBLAB=0, a dummy name can be used in the call.

X Array containing values of independent variable to be plotted.

Y Array containing values of dependent variables to be plotted.

The following informative messages are printed to the line printer by DRAW1:

(a) "XX POINTS DROPPED FROM BEGINNING OF TRACE, NEGATIVE OR ZERO VALUES IN ARRAY"

This message tells the user that there is some number (XX) of invalid values (negative or zero) at the beginning of an array for the plotting of the array on a log scale. The graph will start at the first valid value.

(b) "TRACE CANNOT BE PLOTTED, NEGATIVE OR ZERO VALUES IN ARRAY"

This message is printed when an invalid value (negative or zero) occurred after valid values for the plotting of the array on a log scale. The graph will not be plotted.

(c) "AMIN=XX AMAX=YY"

This message provides the user with additional information on the minimum and maximum values of an array when their difference is very small and expansion of the resolution of the scale would cause the scale notation to overlap illegibly.

2.1.2 Subroutine DRAW4

The FORTRAN callable computer subprogram DRAW4[2] is designed to plot one or more functions (data sets) on the same graph. Any number of data sets can be plotted on each graph, and up to eight graphs can be processed "simultaneously." The axes, scaling, user-defined annotations (axis labels, plot title, and plot subtitle), and the type of plots available (linear-linear, log-log, linear-log, and log-linear) are processed and displayed by the plotter in the same manner as in the subprogram DRAW1 (see sect. 2.1.1). As in DRAW1, the date and time of execution of each graph are displayed in the lower right-hand corner of the graph, and the first time the subprogram is called, a PM message is generated to indicate that "THIS JOB PLOTS." However, DRAW4 has greater capabilities than DRAW1 for the presentation of the data sets and the appearance of the overall plot.

---

[2] *Thomas V. Noon, Enhanced Plotting Software for Use with the Houston Instrument Complot Plotter, Harry Diamond Laboratories TM-75-32 (December 1975).*

To provide the multiple plotting capability, DRAW4 is divided into three major sections. The first section, accessed by setting the first formal parameter (IP) to 1, sets up the user-defined annotations and a scratch disc file identification. This section also rewinds and initializes the scratch file, allowing the file to be reused without user intervention. The second section, accessed by setting IP to 2, processes each data set and the manner the data are to be plotted, and stores this information on the scratch disc file initialized in the first section. The second section must be called for each data set to be plotted and can be called any number of times. The third section, accessed by setting IP to 3, completes the processing of all the information provided in the first and second sections for the specified scratch disc file and sends the information to the OUTPUT file.

This procedure requires N+2 calls to the subprogram DRAW4, where N is the number of data sets to be plotted. Both the first and third sections are called once for each plot. By using this procedure, DRAW4 can process up to eight unique graphs simultaneously, thereby avoiding the need to recalculate information that is to be plotted later along with other information in the program. After closing a graph by a call to DRAW4 with IP=3, the scratch disc file used for that particular graph is available for processing another graph or any other application required in the program.

The first section of DRAW4 (IP=1), as well as establishing the user-defined annotations, declares the scratch disc file to be used for the graph. The user supplies the name and size of the arrays that contain the labels of the independent and dependent axes and the plot title and subtitle. If a value of zero is passed for the size of any one of the above-mentioned arrays, that notation is omitted from the graph. The label information supplied is then stored on the declared scratch disc file for use in the third section.

The second section of DRAW4 (IP=2) receives the data set to be plotted and provides the user with a number of choices for the manner in which a particular data set is displayed on the graph. A data set can be plotted with a solid line, a dashed line, a dotted line, or a dash-dot line. In addition to these line types, all standard characters and various symbols (seven are provided) can be used to mark the occurrence of actual data points in a data set. These symbols can be used by themselves or with one of the line types. The frequency of the plotting of the markers can also be varied (e.g., every data point, every tenth data point, etc). The second section of DRAW4 also allows for the specification of minimum and maximum values for the dependent array, so that various graphs may be compared. If the specified values fall outside the range of the dependent values of all the data sets to be plotted, these specified values are used in the third section of DRAW4 to establish the scale of the dependent axis. If the specified values are inside the range of the dependent arrays, these arrays are changed to conform with the given minimum or maximum. Otherwise, the data arrays passed to this section are not altered.

The third section of DRAW4 (IP=3) allows the user more control over the drawing of the axes, zero reference lines, and border, and over the selection of scale factors and the color of ink used to plot a particular data set. The drawing of the axes and zero reference lines is handled in the same manner as in DRAW1 (see sect. 2.1.1).

The addition of a border to the graph and the use of different colored ink are handled by the same formal parameter. The two options can be selected individually or together. When axes are drawn in the standard position, a border is drawn at the top and right of the graph connected to the major axes. This border consists of axes of the same type (linear or log) as those directly opposite, but without annotation. When the option to change pen color is selected, a PM message is printed on the operator's console after the graph is labeled and the axes drawn, and before each data set is plotted. The PM message puts the output stream in a wait status, the text of the message requests the operator to change the pen on the plotter (the user must have told the operator in advance which colors to use in what order). There is no automatic pen selector with this plotter. When the graph is completed, another PM message requests the restoration of the original color.

Finally, scaling factors are controlled through the third section. These are used to normalize the data set(s) being plotted (this scaling factor is the power of 10 that appears as part of the axis label), to set the increments that determine the scale for a linear axis, and to set the minimum and maximum powers of 10 for a logarithmic axis. These four values must be encoded into one integer word ("INCR" in this software) of twelve digits. Each value is allotted three digits. For linear axes, the factor for the independent axis occupies the high-order (left-most) digits, followed by the increment for the independent scale, the scaling factor for the dependent axis, and the increment for the dependent scale. For logarithmic axes, the minimum power of 10 for the independent axis occupies the high-order digits, followed by the maximum power of 10 for the independent axis, the minimum power of 10 for the dependent axis, and the maximum power of 10 for the dependent axis. The range that the exponents for the scaling factors for the log scale can cover is from +99 to -99. To cover the positive range, the high-order digit of the appropriate set of three digits must be zero. For the negative range, the high-order digit of the appropriate set of three digits must be one. An exponent of zero is achieved by setting the appropriate digits to 200 (setting the digits to zero causes the INCR to be ignored). For a logarithmic scale the uses of a partial decade preceding the minimum value or following the maximum value are not considered if 999 is encoded into the appropriate portion of INCR. If the data set(s) being plotted cannot meet the user-defined scaling factor(s), scale increment(s), or minimum and maximum scale values, the supplied values are ignored and new values are computed. A printed message notifies the user of this action. The values used to define the scaling factors are as follows:

POWX	exponent of 10 (10^{POWX}) to which the independent arrays are normalized
ΔX	scale increment for the independent axis
POWY	exponent of 10 (10^{POWY}) to which the dependent arrays are normalized
ΔY	scale increment for the dependent axis
MINX	minimum power of 10 for the independent axis
MAXX	maximum power of 10 for the independent axis
MINY	minimum power of 10 for the dependent axis
MAXY	maximum power of 10 for the dependent axis

Order of encoding of information:

POWX ΔX POWY ΔY	linear-linear plot
MINX MAXX MINY MAXY	log-log plot
MINX MAXX POWY ΔY	log-linear plot
POWX ΔX MINY MAXY	linear-log plot

Specific examples of the defining of scaling factors follow.

(a) To have the independent array(s) normalized by 10^{-6} and a scale increment of 20, with the dependent array(s) normalized by 10^{3} and a scale increment of 5,

POWX will equal 106 (100 indicates negative power of 10

ΔX will equal 020

POWY will equal 003

ΔY will equal 005

∴ INCR will equal 106020003005

(b) To have the independent array(s) normalized by 10^{0} with no specified scale increment, and the dependent array(s) plotted with DRAW4 software calculated values,

POWX will equal 200 (200 indicates a factor of 1)

ΔX will equal 000

POWY will equal 000

ΔY will equal 000

∴ INCR will equal 200000000000

(c) To have independent array(s) plotted on a logarithmic scale covering from 10^{-1} to 10^{4}, and dependent array(s) plotted on a logarithmic scale covering from 10^{-6} to 10^{-1},

MINX will equal 101

MAXX will equal 004

MINY will equal 106

MAXY will equal 101

∴ INCR will equal 101004106101

(d) To have the independent array(s) plotted on a linear scale normalized by 10^{-6} with no specified scale increment and the dependent array(s) plotted on a logarithmic scale starting at 10^{1} with no specified maximum and no partial decade with the calculated maximum,

POWX will equal 106

ΔX will equal 000

MINY will equal 001

MAXY will equal 999

∴ INCR will equal 106000001999

The values contained in the scratch arrays provided to DRAW4 in the third section (IP=3) are restored if the value for the number of points in the scratch arrays is positive. This is done by storing the values in the scratch arrays on the same scratch disc file used to store previous plot information, and restoring the values to the scratch arrays before returning to the calling program from DRAW4. If the value for the number of points in the scratch arrays is negative, the values for the scratch arrays are not restored.

The FORTRAN subprogram DRAW4 is called as follows:

For IP=1:

```
CALL DRAW4(IP,IFILE,IXLAB,IYLAB,IPTLAB,ISBLAB,XLAB,YLAB,
           PTLAB,SUBLAB)
```

where

IFILE — A number from 1 to 8 identifying the scratch file associated with the graph. For each graph the value must be consistent for each value of IP.

IXLAB — Number of words in the label for the independent axis, not to exceed a value of 3. If IXLAB=0, axis label is omitted.

IYLAB — Number of words in the label for the dependent axis, not to exceed a value of 3. If IYLAB=0, axis label is omitted.

IPTLAB — number of words in the plot title, not to exceed a value of 8. If IPTLAB=0, plot title is omitted.

ISBLAB — Number of words in the plot subtitle, not to exceed a value of 8. If ISBLAB=0, plot subtitle is omitted.

XLAB — Array containing user-defined label for the independent axis, not to exceed 3 words (30 characters). If IXLAB=0, a dummy name can be used in the call.

YLAB — Array containing user-defined label for the dependent axis, not to exceed 3 words (30 characters). If IYLAB=0, a dummy name can be used in the call.

PTLAB — Array containing user-defined title for the plot, not to exceed 8 words (80 characters). If IPTLAB=0, a dummy name can be used in the call.

SUBLAB — Array containing user-defined subtitle for the plot, not to exceed 8 words (80 characters). If ISBLAB=0, a dummy name can be used in the call.

For IP=2:

```
CALL DRAW4(IP,IFILE,ITYPE,NPTS,KODE,ISPACE,XVAL,
      YVAL,YMIN,YMAX)
```

where

IFILE — A number from 1 to 8 identifying the scratch file associated with the graph and a previous call to DRAW4 with IP=1.

ITYPE — Code used to select the type of plot. Read only on the first call with IP=2 for each IFILE.

If ITYPE=1 a linear plot is selected.

=2 a log-log plot is selected.

	=3	a linear-log plot (semilog) with the linear axis for the dependent array) is selected.
	=4	a log-linear plot (semilog with the linear axis for the independent array) is selected.

NPTS — Number of points in both the independent and dependent data arrays to be plotted. For multiple plots, NPTS for each data set does not have to be the same.

KODE — The character to be drawn at data points, connected by a line. If KODE is negative, only the sumbol is drawn.

If KODE=0	no character
=1	a plus (+)
=2	an X
=3	a triangle (△)
=4	a square (□)
=5	an hour glass (⧗)
=6	an up arrow (↑)
=7	a star (*)

Other characters can be drawn by equating KODE to a character (e.g., 2R̸A, where 'A' is the character to be drawn at data points).

ISPACE (a) ISPACE/10 is the rate of occurrence for the drawing of the symbol designated by the variable KODE. If ISPACE/10>0 and KODE=0, intermediate points are disregarded to give the user a simplified plot.

(b) The last digit of ISPACE determines the type of line to be drawn, provided KODE is not negative.

If the last digit of ISPACE is

0 a solid line is drawn ______

1 a dashed line is drawn ------

2 a dotted line is drawn. . . .

3 a dash-dot line is drawn -·-·-·

XVAL Array containing values of independent variables to be plotted.

YVAL Array containing values of dependent variables to be plotted.

YMIN Used in the first call with IP=2 to set the minimum value for the dependent axis. If YMIN is zero (0.) it is ignored. For a minimum value of zero, set YMIN equal to a very small number (e.g., 1.E-30). Any values in the arrays that are less than YMIN are set equal to YMIN, thus modifying the arrays.

YMAX Used in the first call with IP=2 to set the maximum value for the dependent axis. If YMAX is zero (0.) it is ignored. Any values in the arrays that are greater than YMAX are set equal to YMAX, thus modifying the arrays.

For IP=3:

```
CALL DRAW4(IP,IFILE,IDUM,INCR,ICHOICE,N,X,Y,ZAX,ZAY)
```

where

IFILE A number from 1 to 8 identifying the scratch file associated with the graph and previous calls to DRAW4 with IP=1 and 2.

IDUM Dummy variable, presently not used.

INCR Variable used to specify scales other than those calculated by the subprogram. A single integer number composed of four values (POWX, ΔX, POWY, ΔY, MINX, MAXX, MINY, or MAXY) which are used for the normalizing factors and scale increments; ignored if equal to zero (0).

ICHOICE Variable used to request a change in pen color and to draw borders on graph.

If ICHOICE=0 no action is taken.

=1 a change of pen is requested before the plotting of each data set.

=2 plot is drawn with a border (upper and right sides), if possible.

=3 both options 1 and 2 are implemented.

N Number of points in scratch arrays; should be equal to or greater than largest data set being plotted. If N is negative the scratch arrays are not restored to their original values.

X Scratch array used for temporary storage of independent array(s) to be plotted.

Y Scratch array used for temporary storage of dependent array(s) to be plotted.

ZAX Variable that determines where the independent axis is drawn.

If ZAX=0. axis is drawn in the standard location. This is also the default position when the next two conditions cannot be met.

=1. axis is drawn at the zero occurrence of the dependent axis.

=2. axis is drawn at standard location with the addition of a zero reference line placed at the zero occurrence on the dependent axis.

ZAY — Variable that determines where the dependent axis is drawn.

If ZAY=0. axis is drawn in standard location. This is also the default position when the next conditions cannot be met.

=1. axis is drawn at the zero occurrence of the independent axis.

=2. axis is drawn at standard location with the addition of a zero reference line placed at the zero occurrence on the independent axis.

The following messages can be printed to the line printer by DRAW4:

(a) "CALLS OUT OF SEQUENCE ON FILE XX"

This message is printed if the second section of DRAW4 is called before the first section, or the third section called before the first or second sections, for the graph associated with scratch file number XX.

(b) "XX POINTS DROPPED FROM TRACE YY IN FILE ZZ, NEG. OR ZERO VALUE IN ARRAY"

This message notifies the user of XX invalid values (negative or zero) for the plotting of the YYth data set on a log scale for the graph associated with scratch file number ZZ.

(c) "TRACE ON FILE XX AFTER TRACE NO. YY CANNOT BE PLOTTED, NEG. OR ZERO VALUE IN ARRAY"

This message is printed when an invalid value (negative or zero) occurs after valid values in data set YY+1 on a log scale for the graph associated with scratch file number XX.

(d) "AMIN=XX AMAX=YY"

This message provides the user with additional information on the minimum and maximum values of an array when their difference is very small and expansion of the resolution of the scale would cause the scale notation to overlap illegibly.

(e) "IMPROPER SCALE IGNORED"

This message notifies the user that the selected values for the scale could not be used. The program then attempts to calculate a scale close to the selected values.

(f) "ARRAY PASSED IN CALL 3, FILE XX, NOT LARGE ENOUGH"

This message warns the user that the dimension of the scratch arrays provided in the third section associated with file number XX is not as large as that of the data sets being plotted. The results are unpredictable.

2.1.3 Subroutine DRAW2

Subroutine DRAW2[3] is a printer-plot version of DRAW1. The printer-plot routine mimics DRAW1 and is useful for making test runs or plots for which lower resolution is acceptable. The calling parameters for the printer-plot routine are identical to those for DRAW1 and the routine is accessed by replacing the calls to DRAW1 with calls to DRAW2.

2.1.4 Subroutine DRAW3

Subroutine DRAW3[3] is a printer-plot version of DRAW4. The printer-plot routine mimics DRAW4 and is useful for making test runs or plots for which a lower resolution is acceptable. The calling parameters for the printer-plot routine are identical to those for DRAW4 and the routine is accessed by replacing the calls to DRAW4 with calls to DRAW3.

2.1.5 Subroutine CALCMP

Subroutine CALCMP translates calls to "CALCMP" (the California Computer Products, Inc., plot subroutine) into a form compatible with the Houston Instrument plotter. Image scaling and beam control are not possible, and reference to these functions causes a stop condition. To implement this routine the user simply loads from the ANAPAC library before loading from the CALCOMP library.

---

[3] *Egon Marx, Printer Version of Plots Made by an Incremental Plotter, Harry Diamond Laboratories TM-75-33 (December 1975).*

2.2 Fourier Transform Routines

2.2.1 Subroutine FFT

Subroutine FFT performs a forward or an inverse Fourier transform of an equispaced array of N points, where N=2**NPOW. Subroutine FFT makes use of the Cooley-Tukey algorithm, programmed in COMPASS assembly language. The input and output for FFT is passed by the same formal parameter, A. For the inverse transform, the output is obtained by taking the REAL part of the output array. By convention, the sign of the exponent in the integral is negative for the forward transform and positive for the inverse transform.

Subroutine FFT is called as follows:

```
CALL FFT(A,NPOW,N,DX,ISIGN)
```

A	Complex array containing input or output data.
NPOW	Power of 2.
N	Number of points in data array.
DX	Increment for independent array.
ISIGN	Plus or minus 1; value of p desired in transform, $\int g(t)e^{p2\pi ift}dt$.

2.2.2 Subroutine FLAT

Subroutine FLAT[4] performs a forward Fourier transform of an equispaced array by approximating the given function with a piecewise linear function and then using the Cooley-Tukey algorithm to obtain a sampling of the Fourier transform. The input and output for FLAT are passed by the same formal parameter, A.

Subroutine FLAT is called as follows:

```
CALL FLAT(A,N,DX,ISIGN)
```

A	Complex array containing input and output data.
N	Number of points in data array; a power of 2.

[4]*Alfred Brandstein and Egon Marx, Numerical Fourier Transform, Harry Diamond Laboratories TR-1748 (December 1975).*

DX	Increment for independent array.
ISIGN	Plus or minus 1; value of p desired in transform $\int g(t)e^{p2\pi ift}dt$.

2.2.3 Subroutine FLIT

Subroutine FLIT[4] performs an inverse Fourier transform of an equispaced array. The input and output for FLIT are passed by the same formal parameter, A. The input data are placed in the first N/2+1 points. The output is obtained by taking the REAL part of the whole output array.

Subroutine FLIT is called as follows:

```
CALL FLIT(A,N,DX,ISIGN)
```

A	Complex array containing data.
N	Number of points in data array; a power of 2.
DX	Increment for independent array.
ISIGN	Plus or minus 1; value of p desired in transform, $\int g(f)e^{p2\pi ift}dt$.

2.2.4 Subroutines NUFT and INUFT

Subroutines NUFT and INUFT[4] perform a direct and an inverse Fourier transform of a function defined by straight lines connecting the data points. Equispaced arrays and a number of points which is a power of 2 are not a prerequisite of either subroutine.

Subroutine NUFT is called as follows:

```
CALL NUFT(X,Y,N1,N2,OM,IOM,FT,DX,JFLAG)
```

X	Independent array containing time data, input.
Y	Dependent array containing amplitude data, input.

---

[4] *Alfred Brandstein and Egon Marx, Numerical Fourier Transform, Harry Diamond Laboratories TR-1748 (December 1975).*

N1 Number of points in input arrays.

N2 Number of point in output arrays.

OM Independent array containing frequency data, input.

IOM Variable that indicates whether OM array is frequency or circular frequency for input and output.

If IOM=1 circular frequency input and output.

=2 circular frequency input, frequency output.

=3 frequency input, circular frequency output.

=4 frequency input and output.

FT Complex array for transformed data, output.

DX Scratch array, dimensioned to N1.

JFLAG Plus or minus 1, sign of exponent for transform.

Subroutine INUFT is called as follows:

```
CALL INUFT(X,Y,N1,N2,OM,IOM,FT,DX,JFLAG)
```

X Independent array for time data, output.

Y Dependent array for amplitude data, output.

N1 Number of points in output arrays.

N2 Number of points in input arrays.

OM Independent array containing frequency data, input.

IOM Variable that indicates whether OM array is frequency or circular frequency for input and output.

If IOM=1 circular frequency input and output.

=2 circular frequency input, frequency output.

=3 frequency input, circular frequency ouptut.

=4 frequency input and output.

FT Complex array containing transformed data, input.

DX Complex scratch array, dimensioned to N2.

JFLAG Plus or minus 1, sign of exponent of transformed data.

2.2.5 Subroutine FFTA

Subroutine FFTA performs a forward Fourier transform or an inverse Fourier transform with full or half aliasing. For a forward transform, FFTA simply calls FFT with a negative exponent. For half aliasing, the values in the second half of the transform array, on the midpoint plus one, are replaced with the conjugate of the respective value of the first half. For full aliasing, the conjugates of the values in the first half of the transform array, on the midpoint plus one, are added to the respective values in the second half of the transform array. Then the conjugates of the sums for each value in the second half replace the respective values in the first half. Full aliasing should be used only on analytic functions. The result of either the half or the full aliasing is then used as input to FFT. A phenomenon that can occur with inverse FFT's is an overall amplitude shift of the resulting function. To adjust for this shift, the formal parameter ZRO allows the user to shift the total function so that the zero time point has an amplitude of zero.

Subroutine FFTA is called as follows:

```
CALL FFTA(K,FT,N,DX,ISIGN,ZRO,Y)
```

K Variable that selects the type of transform and type of aliasing.

	If K=0	forward Fourier transform selected.
	=1	inverse Fourier transform with full aliasing selected.
	=2	inverse Fourier transform with half aliasing selected.
FT		Complex array of time data input and transform data output.
N		Number of points in input and output arrays.
DX		Increment for independent arrays.
ISIGN		Plus or minus 1, sign of exponent of transformed data.
ZRO		Logical variable used to select option to shift output time function.
	If ZRO=.T.	function shifted so that first point has amplitude of zero.
	=.F.	function not shifted.
Y		Output array containing real part of inverse Fourier transform.

2.2.6 Subroutine SMUZ

The power spectrum from a digitized time-amplitude trace shows a strong oscillating noise component, especially on a logarithmic scale. Subroutine SMUZ can be used to present the output in a more intelligible form. The input is an array of a function given at constant intervals. When two maxima or two minima occur closer than a prescribed number of points (the threshold), the function in between is replaced by an average value between the straight line joining the maxima and that joining the minima, with a tapered beginning and end. The process is repeated until there are no changes in a complete pass. Two difference thresholds can be prescribed for two sectors of the function. The number of passes through the procedure is printed in the output (usually four to six passes). The thresholds have to be chosen in such a way that the unwanted noise is eliminated while the significant extrema remain; a number to start with might be 1/100 of the total number of points. It has been found that the use of this subroutine on the real and imaginary parts of a Fourier transform that subsequently has to be inverted tends to introduce spurious oscillations for late times.[4]

[4] *Alfred Brandstein and Egon Marx, Numerical Fourier Transform, Harry Diamond Laboratories TR-1748 (December 1975).*

Subroutine SMUZ is called as follows:

CALL SMUZ(X,N,M1,M2,FRAC)

X	Ordinates of function being smoothed.
N	Number of points in X array.
M1	Threshold number of points for first part of curve.
M2	Threshold number of points for second part of curve.
FRAC	Fraction of points in first part of curve.

2.2.7 Subroutine OMEGA

Subroutine OMEGA generates a frequency array. Given the maximum frequency desired, WMAX, and the initial delta frequency, DW, OMEGA constructs an unequispaced array with increments in geometric progression. If DW equals zero, an equispaced frequency array is constructed. The parameters WMAX and DW can be in either units of frequency or circular frequency.

Subroutine OMEGA is called as follows:

CALL OMEGA(OM,WMAX,N,DW)

OM	Output frequency array.
WMAX	Maximum frequency desired.
N	Number of frequency points desired.
DW	Initial frequency step.

2.3 Data Preparation and Interpolation Routines

2.3.1 Subroutine CSTOUT

Subroutine CSTOUT checks for time ordering of the independent array and casts out those points not in an ascending time order. If points are encountered with the same time value, the corresponding values in the dependent array are averaged. Messages notify the user how many points were valid and invalid. Subroutine CSTOUT is recommended for use with digitized data.

Subroutine CSTOUT is called as follows:

CALL CSTOUT(X,Y,N)

X Independent array, time.

Y Dependent array, amplitude.

N Number of points in X and Y arrays; if invalid values encountered, new value of N returned.

2.3.2 Subroutines CLINT and LINT

Subroutines CLINT and LINT perform a linear interpolation between the points in the given array. The outputs from CLINT and LINT are equispaced arrays. The output array for CLINT is complex; that for LINT is real. Any points which fall outside the range of the input variable are given a value of zero in the output array.

Subroutines CLINT and LINT are called as follows:

CALL CLINT(X,Y,N,M,DX,YOUT)

or

CALL LINT(X,Y,N,M,DX,YOUT)

X Independent array, time.

Y Dependent array, amplitude.

N Number of points in X and Y arrays.

M Number of points in YOUT array.

DX Increment for independent variable.

YOUT Dependent output array; complex for CLINT.

2.4 File Structuring Routines

The following subroutines provide a compatible data structure for the easy exchange of data between different computer programs.

2.4.1 Subroutine WRTAPE

Subroutine WRTAPE writes data sets onto a local program file which is to be established as a permanent file for access by another program. Multiple data sets can be written on the same file. The user must properly close the file before termination of the program.

Subroutine WRTAPE is called as follows:

```
CALL WRTAPE(NT,X,Y,N,LABEL)
```

NT	File number where data are written.
X	Independent array; for an equispaced array X(1) contains the increment.
Y	Dependent array.
N	Number of points in data arrays; negative value for N indicates an equispaced independent array.
LABEL	Name of data set (eight words).

2.4.2 Subroutine RDTAPE

Subroutine RDTAPE reads data sets from the permanent files created by using WRTAPE. Subroutine RDTAPE checks for the end of information (EOI) and for any irregularities in the data format. This subroutine reads digitized data files.

Subroutine RDTAPE is called as follows:

```
CALL RDTAPE(NT,X,Y,N,LABEL)
```

NT	File number where data are contained.
X	Independent array; for equispaced array X(1) contains the increment.
Y	Dependent array.
N	Number of points in data arrays; if N is negative, independent array is equispaced.
LABEL	Name of data set (eight words).

2.5 Analytic Signal Generating Routines

A model used to represent analytically the response to EMP is a function that vanishes for $t < 0$, defined by

$$f(t) = At/T_1 \qquad 0 \le T \le T_1$$

$$= A\exp[-\xi_1(t - T_1)] \cos\left[\frac{\pi(t - T_1)}{2(T_2 - T_1)}\right] \qquad T_1 \le t \le T_2$$

$$= \alpha + \beta t + \delta t^2 + \gamma t^3 \qquad T_2 \le t \le T_3$$

$$= a\exp[-\xi_2(t - T_3)] \cos[\omega_1(t - T_3)]$$

$$+ (a\xi_2/\omega_2) \exp[-\xi_3(t - T_3)] \sin[\omega_2(t - T_3)]$$

$$T_3 \le t \le T_E$$

$$= 0 \qquad t > T_E$$

The values of A, T_1, T_2, T_3, and a are defined by the user. The user also gives a time T_E to define the end of the "recorded" part of the pulse. The values of ξ_1, ξ_2 and ξ_3 are given through constants S_1, S_2 and S_3 by

$$\xi_1 = \frac{1}{S_1(T_2 - T_1)}$$

$$\xi_2 = \frac{1}{S_2(T_E - T_3)}$$

$$\xi_3 = \frac{1}{S_3(T_E - T_3)},$$

and the circular frequencies are given by

$$\omega_1 = \frac{\pi}{2(T_4 - T_3)}$$

$$\omega_2 = \frac{\pi S_4}{T_E - T_3} .$$

Finally, α, β, γ, and δ are determined by matching the value and the derivative of the function at the ends of the interval.

The Fourier transform of this function is the sum of the Fourier transforms of the components in the different intervals. These components are

$$F_1(\omega) = \frac{A[\exp(-i\omega T_1)\ (i\omega T_1 + 1) - 1]}{T_1\omega^2}$$

$$F_2(\omega) = A\exp(-i\omega T_1) \left[(\xi_1 + i\omega)^2 + \frac{\pi^2}{4(T_2 - T_1)^2}\right]^{-1}$$

$$\times\{\exp[-(\xi_1 + i\omega)(T_2 - T_1)]\ \frac{\pi}{2(T_2 - T_1)} + (\xi_1 + i\omega)\}$$

$$F_3(\omega) = \exp(-i\omega T_3) \left[\frac{a}{-i\omega} + \frac{2\xi + 6\gamma T_3}{i\omega^3} - \frac{6\gamma}{\omega_4}\right] - \exp(i\omega T_2)$$

$$\left[\frac{\gamma}{\omega^2} + \frac{2\delta + 6\gamma T_2}{i\omega^3} - \frac{6\gamma}{\omega^4}\right]$$

$$F_4(\omega) = a\exp(-i\omega T_3)\left[\frac{\xi_2 + i\omega}{(\xi_2 + i\omega)^2 + \omega_1^2}\right.$$

$$+ \frac{\xi_2}{(\xi_3 + i\omega)^2 + \omega_2^2} + \eta\,\frac{\exp\left[-(\xi_2 + i\omega)\left(T_E - T_3\right)\right]^2}{(\xi_2 + i\omega)^2 + \omega_1^2}$$

$$\times\left[-(\xi_2 + i\omega)\cos\omega_1\left(T_E - T_3\right) + \omega_1 \sin\omega_1\left(T_E - T_3\right)\right]$$

$$+ \eta\,\frac{\xi_2}{\omega_2}\,\frac{\exp\left[-(\xi_3 + i\omega)\left(T_E - T_3\right)\right]}{(\xi_3 + i\omega)^2 + \omega_2^2}$$

$$\left.\times\left[-(\xi_3 + i\omega)\sin\omega_2\left(T_E - T_3\right) - \omega_2\cos\omega_2\left(T_E - T_3\right)\right]\right],$$

where

$$Y = -\frac{\pi A}{2(T_2 - T_1)}\exp[-\xi_1(T_2 - T_1)],$$

$$\gamma = -\frac{2a}{(T_3 - T_2)^2} + \frac{Y}{(T_3 - T_2)^3},$$

$$\delta = -\frac{a}{(T_3 - T_2)^2} - \gamma(T_2 + 2T_3),$$

and η is 1 or 0 depending on whether we assume that the trace vanishes after T_E or is given by the expression in the last interval.

Functions F_1 and F_3 have to be computed separately for $\omega = 0$. They become

$$F_1(0) = \frac{AT_1}{2}$$

$$F_3(0) = \alpha(T_3 - T_2) + \frac{\beta\left(T_3{}^2 - T_2{}^2\right)}{2} + \frac{\left(T_3{}^2 - T_2{}^3\right)}{3} + \frac{\delta\left(T_3{}^4 - T_2{}^4\right)}{4},$$

where

$$\beta = -2T_3\delta - 3T_3{}^2\gamma ,$$

$$\alpha = -T_2\beta - T_2{}^2\delta - T_2{}^3\gamma.$$

The function is continuous at the extremes of the intervals, but it does not necessarily vanish at T_E. The derivative is discontinuous at T_1, but continuous at T_2 (where the function vanishes) and at T_3 (where the function reaches a minimum given by a). At T_4, the first term in the corresponding expression for f(t) vanishes.

2.5.1 Subroutine ANMI

Subroutine ANMI[4] computes the value of f(t) for the analytical model described in section 2.5. The desired time points and the number of points and input to ANMI and the amplitude values are returned. Subroutine ANMI requires a NAMELIST card, PULSE, that contains the values A, AA, T1, T2, T3, T4, TE, S1, S2, S3, and S4. The meanings of these are evident, except that AA = a.

Subroutine ANMI is called as follows:

```
CALL ANMI(X,Y,N)
```

X Time points at which function evaluated.

Y Array for amplitudes.

N Number of points in X.

---

[4] *Alfred Brandstein and Egon Marx, Numerical Fourier Transform, Harry Diamond Laboratories TR-1748 (December 1975).*

2.5.2 Subroutine ANTRA

Subroutine ANTRA[4] computes the Fourier transform for the analytical model described in section 2.5. The desired frequency points (circular frequency) and the number of points are input to ANTRA and the values of the Fourier transform are returned. Subroutine ANTRA requires a NAMELIST card, PULSE, which contains the values A, AA, T1, T2, T3, T4, TE, S1, S2, S3, S4, and FINIT. Variable FINIT takes the value 0. for an infinite trace and 1. for a finite trace.

Subroutine ANTRA is called as follows:

```
CALL ANTRA(OM,FT,N)
```

OM Circular frequencies at which function evaluated.

FT Complex array containing values of the Fourier transform.

N Number of points in OM.

3. THE TRANL USER'S LIBRARY AND SUPPORTING PROGRAMS

The TRANL user's library is a collection of independent computer programs with supporting subroutines which provides the means for the validation of digitized data sets and the means to easily incorporate digitized data into the normal program input structure. The *library* comprises three principal programs. The main program, TRANS, translates the punched card output from the digitizing system, checks for format errors in the data set, provides a time-amplitude listing of the digitized data, and creates a local file of the digitized data which is formatted to comply with subroutines WRTAPE and RDTAPE in the ANAPAC library. If errors are encountered during the translation of a digitized data set, the user is informed of the card number where the error occurred. To aid in the correction of the error the program GRAFC can be used to translate the digitized data set into the control format used in the digitization process. Another supporting program, TEDD, provides for "time window expansion of digitized data." These programs were stored in TRANL with an access level of one, which allows access to the programs by control cards.

Another program associated with the TRANL library, but not stored in it, is REDUCE. Program REDUCE provides the user with the ability to selectively reduce, by defining time windows, the number of points in a digitized data set and create a new data file.

---

[4] *Alfred Brandstein and Egon Marx, Numerical Fourier Transform, Harry Diamond Laboratories TR-1748 (December 1975).*

3.1 Program TRANS

Program TRANS translates the punched card output from the digitizing system into time-amplitude information. The time-amplitude information is listed on the printer and written on a local file, which can be catalogued as a permanent file. The printed listing can be eliminated by putting sense switch 1 to "on." The local or permanent file can then be accessed by the use of RDTAPE from ANAPAC. The input for TRANS can be one or more data sets each preceded by a label card and followed by an end-of-information card (EOI, 7/8/9 punch). The input stream is terminated by two EOI cards.

Program TRANS is accessed by control cards as follows:

```
TRANS(lfn1,lfn2,lfn3)
```

lfn1 Logical file name for input (e.g., INPUT).

lfn2 Logical file name for output (e.g., OUTPUT).

lfn3 Logical file name assigned by user to receive time-amplitude data.

3.2 Program GRAFC

Program GRAFC converts the punched-card digitized data into the control commands and data points recorded by the digitizer. The output listing from GRAFC greatly aids in error corrections and modification of digitized data and can eliminate the need for re-digitization. The input to GRAFC is the same as for TRANS.

Program GRAFC is accessed by control cards as follows:

```
GRAFC (lfn1,lfn2)
```

lfn1 Logical file name for input (e.g., INPUT).

lfn2 Logical file name for output (e.g., OUTPUT).

3.3 Program TEDD

Program TEDD provides the user with the ability to plot selected time windows of a digitized data set. The user can thus expand portions of a time-amplitude trace for clarity or display the individual digitized points in the selected time window. By displaying individual digitized points, invalid points can be identified and either be removed from the data deck or have their values changed. To aid in the correction of a digitized trace, TEDD prints the following information about a data set:

(a) The label assigned to the data set

(b) The number of digitized points in the trace

(c) The duration of the trace

(d) The selected boundary points for the time window

(e) The time of the end points in the digitized trace which bound the selected time window and define the time scale for the plotted output

(f) The number of digitized points in the time window

(g) The numbered location of the digitized end points relative to the beginning of the trace

Program TEDD can process multiple data sets with a maximum of six time windows per data set (the time windows can overlap). The maximum and minimum for the dependent variable can also be specified to maintain the same perspective for the plots. The format for the maximum and minimum values and the time windows is '6X,2E10.3' with one time window per card. Each group of time windows must be separated by an EOI card (7/8/9 punch). By putting sense switch 2 to "on," the marking of the individual data points is eliminated.

The input data for TEDD are as follows:

Card 1	AMIN	Minimum value of scale for plotting of dependent array; ignored if equal to zero.
	AMAX	Maximum value of scale for plotting of dependent array; ignored if equal to zero.
		Format (6X,2E10.3)
Card 2	T1	Lower limit of first time window to be expanded.
	T2	Upper limit of first time window to be expanded.
		Format (6X,2E10.3)

Card 3	T1&T2	Limits for second time window; same as Card 2.
.		
.		
.		
Card 7	T1&T2	Limits for sixth time window; same as Card 2.

Program TEDD can also be applied to other types of data files compatible with RDTAPE from the ANAPAC library. Program TEDD requires the support of the ANAPAC library.

Program TEDD is accessed by control cards as follows:

TEDD(lfn1,lfn2,lfn3)

lfn1 Logical file name for input (e.g., INPUT).

lfn2 Logical file name for output (e.g., OUTPUT).

lfn3 Logical file name assigned by user which contains data to be examined.

3.4 Program REDUCE

Program REDUCE enables the user to selectively reduce the number of points in a digitized record. Program REDUCE operates in two modes. The first mode allows the user to define from one to four time windows and the percentage of data points to be saved from each time window. In the second mode, the user provides the factor by which the trace is to be reduced. The reduced time trace is then written on a user-defined logical file using WRTAPE and also plotted to display the results of the reduction. Program REDUCE requires the support of the ANAPAC library.

The input data for REDUCE are as follows:

Card 1	Factor	Factor by which data set is to be reduced; if 0, time windows and percentages defined on following cards.
	M	Number of points to which data set is to be reduced; can be omitted if Factor≠0.
		Format (I2,2X,I4)

Card 2 (if Factor=0)	Ti	Upper limit of time window; i=1,2,3; last time window bounded by Ti and maximum time of data set if $\Sigma Pi \neq 100$ percent. Format (3E10.3)
Card 3 (if Factor=0)	Pi	Percentage of points selected from corresponding time windows of i; i=1,2,3; for each Ti there must be a Pi. Format (3E10.3)

Abnormal termination occurs under the following conditions:

STOP 1 First percentage value equal to zero.

STOP 2 Ti exceeds the maximum time value of the data set.

Program REDUCE is stored as a permanent file and is accessed by a Scope ATTACH card. Loading and execution are accomplished with a Scope file name call. Access is accomplished as follows:

```
ATTACH(A,REDUCE,ID=*****,MR=1)

A(,,lfn1,lfn2)
```

lfn1 Logical file name assigned by user which contains data set to be reduced.

lfn2 Logical file name assigned by user which receives reduced data set.

4. USE OF THE ANAPAC AND TRANL LIBRARIES BY GAMPHS

The computer analysis section of the GAMPHS[1] application (see fig. 3) makes use of the TRANL user's library and many of the subroutines from the ANAPAC user's library. The routines in the TRANL user's library (see sect. 3) and the digitizing system that the routines support provide the link between system field tests and the computer analysis of the field test data. The output of the TRANL routines is a file of digital data. The ANAPAC user library (see sect. 2) provides many of the analytic routines required to analyze the test data. The modular ANAPAC routines are used as building blocks to develop programs which yield meaningful information about the data (e.g., Fourier transform, transfer functions). The routines also prepare the data to serve as input for further analysis by other computer programs. The file structuring routines contained in the ANAPAC user's library (see sect. 2.4) provide the file compatibility necessary to pass data between computer programs.

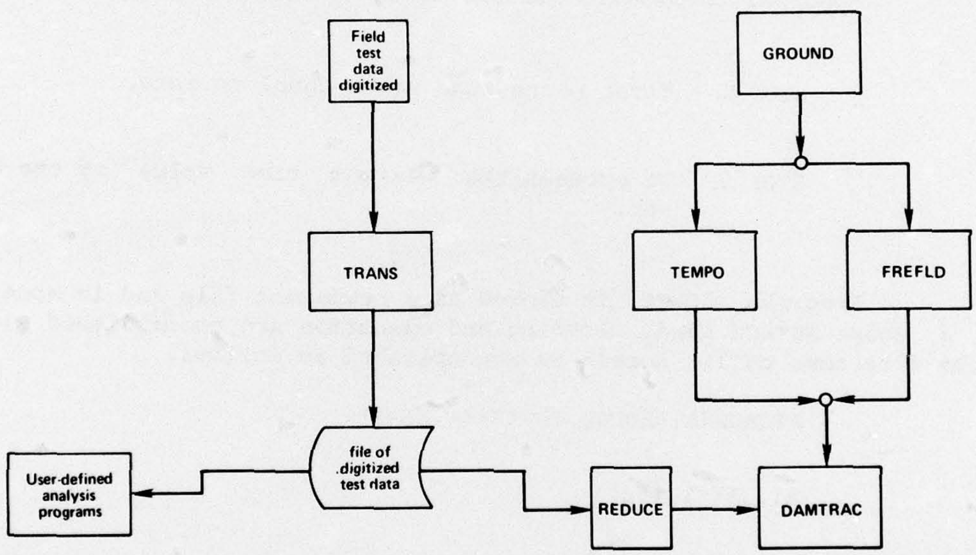

Figure 3. Interaction of computer programs used by GAMPHS.

[1]*George Gornak et al, EMP Assessment for Army Tactical Communications Systems: Transmission Systems Series No. 1--Radio Terminal Set AN/TRC-145 (U), Harry Diamond Laboratories TR-1746 (February 1976). (SECRET-RESTRICTED DATA)*

The following are other computer programs which are part of GAMPHS and use portions of the TRANL and ANAPAC user's libraries for analytic support and file compatability:

1. GROUND:[5] Determines the total electric and magnetic fields in air and those transmitted below the surface for a plane-wave pulse incident on a ground of finite conductivity. The output can be used as input for TEMPO or FREFLD. Uses Fourier transform, plotting and file structuring routines.

2. TEMPO:[1] Computes the transient response of a cable, antenna, or aperture to EMP stimulation; both a batch and interactive code. The output can be used as input for DAMTRAC. Uses plotting and file structuring routines.

3. FREFLD:[6] Calculates the current response of shielded and unshielded cables to EMP stimulation. The output can be used as input for DAMTRAC. Uses Fourier transform, plotting and file structuring routines.

4. DAMTRAC:[7] Performs damage analysis of modeled solid-state circuits from EMP stimulation. Uses data-preparation, plotting and file-structuring routines. See appendix A of this report for the use of digitized data by DAMTRAC.

---

[1]*George Gornak et al, EMP Assessment for Army Tactical Communications Systems: Transmission Systems Series No. 1--Radio Terminal Set AN/TRC-145 (U), Harry Diamond Laboratories TR-1746 (February 1976). (SECRET-RESTRICTED DATA)*

[5]*Egon Marx, Reflected and Transmitted Fields for a Plane-Wave Pulse Incident on a Conducting Ground, Harry Diamond Laboratories TR-1740 (April 1975).*

[6]*Robert F. Gray, Nuclear Electromagnetic Pulse Simulation by Point Source Injection Technique for Shielded and Unshielded Penetration, Harry Diamond Laboratories TR-1737 (December 1975).*

[7]*George Baker et al, Damage Analysis Modified TRAC, Harry Diamond Laboratories TM-76-6 (May 1975).*

LITERATURE CITED

(1) George Gornak et al, EMP Assessment for Army Tactical Communications Systems: Transmission Systems Series No. 1--Radio Terminal Set AN/TRC-145 (U), Harry Diamond Laboratories TR-1746 (February 1976). (SECRET-RESTRICTED DATA)

(2) Thomas V. Noon, Enhanced Plotting Software for Use with the Houston Instrument Complot Plotter, Harry Diamond Laboratories TM-75-32 (December 1975).

(3) Egon Marx, Printer Version of Plots Made by an Incremental Plotter, Harry Diamond Laboratories TM-75-33 (December 1975).

(4) Alfred Brandstein and Egon Marx, Numerical Fourier Transform, Harry Diamond Laboratories TR-1748 (December 1975).

(5) Egon Marx, Reflected and Transmitted Fields for a Plane-Wave Pulse Incident on a Conducting Ground, Harry Diamond Laboratories TR-1740 (April 1975).

(6) Robert F. Gray, Nuclear Electromagnetic Pulse Simulation by Point Source Injection Technique for Shielded and Unshielded Penetration, Harry Diamond Laboratories TR-1737 (December 1975).

(7) George Baker et al, Damage Analysis Modified TRAC, Harry Diamond Laboratories TM-76-6 (May 1975).

APPENDIX A.--COMPATIBILITY OF DAMTRAC WITH DIGITIZED DATA FILES AND THE PROPER APPLICATION OF THE DATA

To enhance user ability to describe transient circuit driving functions, DAMTRAC[1] has been designed to accept time-domain data files structured for WRTAPE/RDTAPE compatibility (see sect. 2.4 of the body of the report). Time-domain data files compatible with the WRTAPE/RDTAPE format include digitized data files and data files generated by the programs covered by the GAMPHS application (see sect. 4 of the body of the report). During a circuit analysis run, the digitized or foreign data file is "attached" and copied to logical unit 7. The DAMTRAC subroutine DGSRCE reads the time-amplitude data from logical unit 7 and edits the data, reducing the number of points to 500, if the input data exceed 500. However, the editing scheme used by DGSRCE is to delete every other point for files containing between 500 and 1000 points and to delete every third point for files containing more than 1000 points. This method of reducing the number of points used for the analysis can inadvertently alter the signature of the transient driving function. For this reason the use of the program REDUCE (see sect. 3.4 of the body of the report), which selectively reduces the number of time-amplitude points and displays the results, is recommended for the reduction of driving function inputs to DAMTRAC.

---

[1]*George Baker et al, Damage Analysis Modified TRAC, Harry Diamond Laboratories TM-76-6 (May 1975).*

DISTRIBUTION

DEFENSE DOCUMENTATION CENTER
CAMERON STATION, BUILDING 5
ALEXANDRIA, VA 22314
ATTN DDC-TCA (12 COPIES)

COMMANDER
USA RSCH & STD GP (EUR)
BOX 65
FPO NEW YORK 09510
ATTN LTC JAMES M. KENNEDY, JR.
CHIEF, PHYSICS & MATH BRANCH

COMMANDER
US ARMY MATERIEL DEVELOPMENT
& READINESS COMMAND
5001 EISENHOWER AVENUE
ALEXANDRIA, VA 22333
ATTN DRXAM-TL, HQ TECH LIBRARY
ATTN DRCRD-WM
ATTN DRCRP, MG C. M. MCKEEN, JR.
ATTN DRCRP-M, COL R. W. SPECKER
ATTN DRCPM-SCM-WF,
LTC D. D. GILPATRICK
ATTN DRCRD-U, COL J. F. BLEECKER
ATTN DRCRD, DIR RES, DEV, & ENGR
ATTN DRCRD-SI, H. DARRICOTT
ATTN DRCSO, SURVEILLANCE, TARGET ACQ
ATTN DRCMS-I, DR. R. P. UHLIG
ATTN DRCMS-I, MR. E. O'DONNEL

COMMANDER
USA ARMAMENT COMMAND
ROCK ISLAND, IL 61201
ATTN DRSAR-ASF, FUZE DIV
ATTN DRSAR-RDF, SYS DEV DIV - FUZES
ATTN DRSAR-PDM, J. A. BRINKMAN
ATTN DRCPM-VFF

COMMANDER
USA MISSILE & MUNITIONS
CENTER & SCHOOL
REDSTONE ARSENAL, AL 35809
ATTN ATSK-CTD-F

DEFENSE ADVANCED RESEARCH
PROJECTS AGENCY
1400 WILSON BLVD
ARLINGTON, VA 22209
ATTN TECH INFORMATION OFFICE
ATTN DIR, STRATEGIC TECHNOLOGY
ATTN DIR, TACTICAL TECHNOLOGY

DIRECTOR
DEFENSE COMMUNICATION ENG CENTER
1860 WIEHLE AVENUE
RESTON, VA 22090
ATTN R104, M. J. RAFFENSPERGER
ATTN R800, R. E. LYONS

DIRECTOR
DEFENSE INTELLIGENCE AGENCY
WASHINGTON, DC 20301
ATTN DI-2, WEAPONS & SYSTEMS DIV

DIRECTOR
DEFENSE NUCLEAR AGENCY
WASHINGTON, DC 20305
ATTN PETER HAAS, DEP DIR,
SCIENTIFIC TECHNOLOGY
ATTN RAEV, MAJ S. O. KENNEDY, SR.
ATTN VLIS, LTC SHIMERDA

DEPARTMENT OF DEFENSE
DIRECTOR OF DEFENSE RESEARCH & ENGINEERING
WASHINGTON, DC 20301
ATTN DEP DIR (TACTICAL WARFARE PROGRAMS)
ATTN DEP DIR (TEST & EVALUATION)
ATTN DEFENSE SCIENCE BOARD
ATTN ASST DIR SALT SUPPORT GP/MR. J. BLAYLOCK

CHAIRMAN
JOINT CHIEFS OF STAFF
WASHINGTON, DC 20301
ATTN J-3, NUCLEAR WEAPONS BR
ATTN J-3, EXER PLANS & ANALYSIS DIV
ATTN J-5, NUCLEAR DIR NUCLEAR POLICY BR
ATTN J-5, REQUIREMENT & DEV BR
ATTN J-6, COMMUNICATIONS-ELECTRONICS

DEPARTMENT OF DEFENSE
JOINT CHIEFS OF STAFF
STUDIES ANALYSIS & GAMING AGENCY
WASHINGTON, DC 20301
ATTN STRATEGIC FORCES DIV
ATTN GEN PURPOSE FORCES DIV
ATTN TAC NUC BR
ATTN SYS SUPPORT BR

ASSISTANT SECRETARY OF DEFENSE
PROGRAM ANALYSIS AND EVALUATION
WASHINGTON, DC 20301
ATTN DEP ASST SECY (GEN PURPOSE PROG)
ATTN DEP ASST SECY (REGIONAL PROGRAMS)
ATTN DEP ASST SECY (RESOURCE ANALYSIS)

DEPARTMENT OF THE ARMY
OFFICE, SECRETARY OF THE ARMY
WASHINGTON, DC 20301
ATTN ASST SECRETARY OF THE ARMY (I&L)
ATTN DEP FOR MATERIEL ACQUISITION
ATTN ASST SECRETARY OF THE ARMY (R&D)

DEPARTMENT OF THE ARMY
ASSISTANT CHIEF OF STAFF FOR INTELLIGENCE
WASHINGTON, DC 20301
ATTN DAMI-OC/COL J. A. DODDS
ATTN DAMI-TA/COL F. M. GILBERT

US ARMY SECURITY AGENCY
ARLINGTON HALL STATION
4000 ARLINGTON BLVD
ARLINGTON, VA 22212
ATTN DEP CH OF STAFF RESEARCH & DEVELOPMENT

DEPARTMENT OF THE ARMY
US ARMY CONCEPTS ANALYSIS AGENCY
8120 WOODMONT AVENUE
BETHESDA, MD 20014
ATTN COMPUTER SUPPORT DIV
ATTN WAR GAMING DIRECTORATE
ATTN METHODOLOGY AND RESOURCES DIRECTORATE
ATTN SYS INTEGRATION ANALYSIS DIRECTORATE
ATTN JOINT AND STRATEGIC FORCES DIRECTORATE
ATTN FORCE CONCEPTS AND DESIGN DIRECTORATE
ATTN OPERATIONAL TEST AND EVALUATION AGENCY

DIRECTOR
NATIONAL SECURITY AGENCY
FORT GEORGE G. MEADE, MD 20755

COMMANDER-IN-CHIEF
EUROPEAN COMMAND
APO NEW YORK, NY 09128

HEADQUARTERS
US EUROPEAN COMMAND
APO NEW YORK, NY 09055

DIRECTOR
WEAPONS SYSTEMS EVALUATION GROUP
OFFICE, SECRETARY OF DEFENSE
400 ARMY-NAVY DRIVE
WASHINGTON, DC 20305
ATTN DIR, LT GEN GLENN A. KENT

DEPARTMENT OF THE ARMY
DEPUTY CHIEF OF STAFF FOR OPERATIONS & PLANS
WASHINGTON, DC 20301
ATTN DAMO-RQZ/LTC L. A. WEIZEL
ATTN DAMO-RQD/COL E. W. SHARP
ATTN DAMO-SSP/COL D. K. LYON
ATTN DAMO-SSN/LTC R. E. LEARD
ATTN DAMO-SSN/LTC B. C. ROBINSON
ATTN DAMO-RQZ/COL G. A. POLLIN, JR.
ATTN DAMO-TCZ/MG T. M. RIENZI
ATTN DAMO-ZD/A. GOLUB

DEPARTMENT OF THE ARMY
CHIEF OF RESEARCH DEVELOPMENT AND ACQUISITION OFFICE
WASHINGTON, DC 20301
ATTN DAMA-RAZ-A/R. J. TRAINOR
ATTN DAMA-CSM-N/LTC OGDEN

CHIEF OF RES DEV & ACQ OFFICE (Cont'd)
ATTN DAMA-WSA/COL W. E. CROUCH, JR.
ATTN DAMA-WSW/COL L. R. BAUMANN
ATTN DAMA-CSC/COL H. C. JELINEK
ATTN DAMA-CSM/COL H. R. BAILEY
ATTN DAMA-WSZ-A/MG D. R. KEITH
ATTN DAMA-WSM/COL J. B. OBLINGER, JR.
ATTN DAMA-PPR/COL D. E. KENNEY

COMMANDER
BALLISTIC MISSILE DEFENSE SYSTEMS
P.O. BOX 1500
HUNTSVILLE, AL 35807
ATTN BMDSC-TEN/MR. JOHN VEFNEMAN

COMMANDER
US ARMY FOREIGN SCIENCE AND TECHNOLOGY CENTER
220 SEVENTH ST., NE
CHARLOTTESVILLE, VA 22901

DIRECTOR
US ARMY MATERIEL SYSTEMS ANALYSES ACTIVITY
ABERDEEN PROVING GROUND, MD 21005
ATTN DRXSY-C/DON R. BARTHEL
ATTN DRXSY-T/P. REID

COMMANDER
US ARMY SATELLITE COMMUNICATIONS AGENCY
FT. MONMOUTH, NJ 07703
ATTN LTC HOSMER

DIRECTOR
BALLISTIC RESEARCH LABORATORIES
ABERDEEN PROVING GROUND, MD 21005
ATTN DRXBR-XA/MR. J. MESZARDS

COMMANDER
US ARMY AVIATION SYSTEMS COMMAND
12TH AND SPRUCE STREETS
ST. LOUIS, MO 63160
ATTN DRCPM-AAH/ROBERT HUBBARD

DIRECTOR
EUSTIS DIRECTORATE
US ARMY AIR MOBILITY R&D LABORATORY
FORT EUSTIS, VA 23604
ATTN SAVDL-EU-MOS/MR. S. POCILUYKO
ATTN SAVDL-EU-TAS(TETRACORE)

COMMANDER
2D BDE, 101ST ABN DIV (AASLT)
FORT CAMPBELL, KY 42223
ATTN AFZB-KB-SO/CPT PAUL C. SMITH

COMMANDER
US ARMY ELECTRONICS COMMAND
FT. MONMOUTH, NJ 07703
ATTN PM, ATACS/DRCPM-ATC/LTC DOBBINS
ATTN DRCPM-ATC-TM
ATTN PM, ARTADS/DRCPM-TDS/BG A. CRAWFORD

US ARMY ELECTRONICS COMMAND (Cont'd)
ATTN DRCPM-TDS-TF/COL D. EMERSON
ATTN DRCPM-TDS-TO
ATTN DRCPM-TDS-FB/LTC A. KIRKPATRICK
ATTN PM, MALOR/DRCPM-MALR/COL W. HARRISON
ATTN PM, NAVCOM/DRCPM-NC/
COL C. MCDOWELL, JR.
ATTN PM, REMBASS/DRCPM-RBS/
COL R. COTTEY, SR.
ATTN DRSEL-TL-IR/MR. R. FREIBERG
ATTN DRSEL-SA/NORMAN MILLSTEIN
ATTN DRSEL-MA-C/J. REAVIS

COMMANDER
US ARMY MISSILE COMMAND
REDSTONE ARSENAL, AL 35809
ATTN DRSMI-FRR/DR. F. GIPSON
ATTN DRCPM-HA/COL P. RODDY
ATTN DRCPM-LCCX/L. B. SEGGEL (LANCE)
ATTN DRCPM-MD/GENE ASHLEY (SAM-D)
ATTN DRCPM-MP
ATTN DRCPM-PE/COL SKEMP (PERSHING)
ATTN DRCPM-SHO
ATTN DRCPM-TO
ATTN DRSMI-R, RDE + MSL DIRECTORATE

COMMANDER
PICATINNY ARSENAL
DOVER NJ 07801
ATTN SARPA-ND-V/DANIEL WAXLER

COMMANDER
US ARMY TANK/AUTOMOTIVE COMMAND
WARREN, MI 48090
ATTN DRSI-RHT/MR. P. HASEK
ATTN DRCPM(XM-L)/MR. L. WOOLCOT
ATTN DRCPM-GCM-SW/MR. R. SLAUGHTER

PRESIDENT
DA, HA, US ARMY ARMOR AND ENGINEER BOARD
FORT KNOX, KY 40121
ATTN STEBB-MO/MAJ SANZOTERRA

COMMANDER
WHITE SANDS MISSILE RANGE
WHITE SANDS MISSILE RANGE, NM 88002
ATTN STEWS-TE-NT/MARVIN SQUIRES

COMMANDER
TRASANA
SYSTEM ANALYSIS ACTIVITY
WHITE SANDS, NM 88002
ATTN ATAA-TDO/DR. D. COLLIER

COMMANDER
197TH INFANTRY BRIGADE
FORT BENNING, GA 31905
ATTN COL WASIAK

COMMANDER
US ARMY COMMUNICATIONS COMMAND
FORT HUACHUCA, AZ 85613
ATTN ACC-AD-C/H. LASITTER (EMP STUDY GP)

COMMANDER
USA COMBINED ARMS COMBAT DEVELOPMENTS
ACTIVITY
FT. LEAVENWORTH, KS 66027
ATTN ATCAC
ATTN ATCACO-SD/LTC L. PACHA
ATTN ATCA/COC/COL HUBBERT
ATTN ATCA-CCM-F/LTC BECKER
ATTN ATSW-TA-3 NUCLEAR STUDY
TEAM/LT D. WILKINS

PROJECT MANAGER
MOBILE ELECTRIC POWER
7500 BACKLICK ROAD
SPRINGFIELD, VA 22150
ATTN DRCPM-MEP

COMMANDER
US ARMY NUCLEAR AGENCY
FT. BLISS, TX 79916
ATTN ATCN-W/COL A. DEVERILL

COMMANDER
US ARMY SIGNAL SCHOOL
FT. GORDON, GA 30905
ATTN AISO-CID/BILL MANNELL
ATTN ATST-CTD-CS/CAPT G. ALEXANDER (INTACS)
ATTN ATSO-CID-CS/LTC R. LONGSHORE

DIRECTOR
JOINT TACTICAL COMMUNICATIONS OFFICE
FT. MONMOUTH, NJ 07703
ATTN TRI-TAC/NORM BECHTOLD

COMMANDER
US ARMY COMMAND AND GENERAL STAFF COLLEGE
FORT LEAVENWORTH, KS 66027

COMMANDER
US ARMY COMBAT DEVELOPMENTS EXPERIMENTATION
COMMAND
FORT ORD, CA 93941

COMMANDER
HQ MASSTER
FORT HOOD, TX 76544

COMMANDER
US ARMY AIR DEFENSE SCHOOL
FORT BLISS, TX 79916
ATTN ATSA-CD

COMMANDER
US ARMY ARMOR SCHOOL
FORT KNOX, KY 40121
ATTN ATSB-CTD

COMMANDER
US ARMY AVIATION CENTER
FORT RUCKER, AL 36360
ATTN ATST-D-MS

COMMANDER
US ARMY ORDNANCE CENTER AND SCHOOL
ABERDEEN PROVING GROUND, MD 21005
ATTN USAOC&S
ATTN ATSL-CTD

COMMANDER
US ARMY SIGNAL SCHOOL
FORT GORDON, GA 30905
ATTN ATSS-CTD

COMMANDER
US ARMY ENGINEER SCHOOL
FORT BELVOIR, VA 22060
ATTN ATSE-CTD

COMMANDER
US ARMY INFANTRY SCHOOL
FORT BENNING, GA 31905
ATTN ATSH-CTD

COMMANDER
US ARMY INTELLIGENCE CENTER
AND SCHOOL
FORT HUACHUCA, AZ 85613
ATTN ATSI-CTD

COMMANDER
US ARMY FIELD ARTILLERY SCHOOL
FORT SILL, OK 73503
ATTN ATSF-CTD

CHIEF OF NAVAL OPERATIONS
NAVY DEPARTMENT
WASHINGTON, DC 20350
ATTN NOP-932, SYS EFFECTIVENESS DIV
CAPT E. V. LANEY
ATTN NOP-9860, COMMUNICATIONS BR
COR L. LAYMAN
ATTN NOP-351, SURFACE WEAPONS BR
CAPT G. A. MITCHELL
ATTN NOP-622C, ASST FOR NUCLEAR
VULNERABILITY, R. PIACESI

COMMANDER
NAVAL ELECTRONICS SYSTEMS COMMAND, HQ
2511 JEFFERSON DAVIS HIGHWAY
ARLINGTON, VA 20360
ATTN PME-117-21, SANGUINE DIV

HEADQUARTERS, NAVAL MATERIEL COMMAND
STRATEGIC SYSTEMS PROJECTS OFFICE
1931 JEFFERSON DAVIS HIGHWAY
ARLINGTON, VA 20390
ATTN NSP2201, LAUNCHING & HANDLING
BRANCH, BR ENGINEER, P. R. FAUROT
ATTN NSP-230, FIRE CONTROL & GUIDANCE
BRANCH, BR ENGINEER, D. GOLD
ATTN NSP-2701, MISSILE BRANCH,
BR ENGINEER, J. W. PITSENBERGER

COMMANDER
NAVAL SURFACE WEAPONS CENTER
WHITE OAK, MD 20910
ATTN CODE 222, ELECTRONICS & ELECTRO-
MAGNETICS DIV
ATTN CODE 431, ADVANCED ENGR DIV

US AIR FORCE, HEADQUARTERS
DCS, RESEARCH & DEVELOPMENT
WASHINGTON, DC 20330
ATTN DIR OF OPERATIONAL REQUIREMENTS
AND DEVELOPMENT PLANS, S/V &
LTC P. T. DUESBERRY

COMMANDER
AF WEAPONS LABORATORY, AFSC
KIRTLAND AFB, NM 87117
ATTN ES, ELECTRONICS DIVISION
ATTN EL, J. DARRAH
ATTN TECHNICAL LIBARY
ATTN D. I. LAWRY

COMMANDER
AERONAUTICAL SYSTEMS DIVISION, AFSC
WRIGHT-PATTERSON AFB, OH 45433
ATTN ASD/YH, DEPUTY FOR B-1

COMMANDER
HQ SPACE AND MISSILE SYSTEMS ORGANIZATION
P.O. 96960 WORLDWAYS POSTAL CENTER
LOS ANGELES, CA 90009
ATTN S7H, DEFENSE SYSTEMS APL SPO
ATTN XRT, STRATEGIC SYSTEMS DIV
ATTN SYS, SURVIVABILITY OFC

SPACE AND MISSILE SYSTEMS ORGANIZATION
NORTON AFB, CA 92409
ATTN MMH, HARD ROCK SILO DEVELOPMENT

COMMANDER
AF SPECIAL WEAPONS CENTER, AFSC
KIRTLAND AFB, NM 87117

HARRY DIAMOND LABORATORIES
ATTN MCGREGOR, THOMAS, COL, COMMANDING
OFFICER/FLYER, I.N./LANDIS, P.E./
SOMMER, H./CONRAD, E.E.
ATTN CARTER, W.W., DR., ACTING TECHNICAL
DIRECTOR/MARCUS, S.M.
ATTN KIMMEL, S., IO
ATTN CHIEF, 0021
ATTN CHIEF, 0022
ATTN CHIEF, LAB 100
ATTN CHIEF, LAB 200
ATTN CHIEF, LAB 300
ATTN CHIEF, LAB 400
ATTN CHIEF, LAB 500
ATTN CHIEF, LAB 600
ATTN CHIEF, DIV 700
ATTN CHIEF, DIV 800

DISTRIBUTION (Cont'd)

HARRY DIAMOND LABORATORIES (Cont'd)
ATTN CHIEF, LAB 900
ATTN RECORD COPY, BR 041
ATTN HDL LIBRARY (3 COPIES)
ATTN CHAIRMAN, EDITORIAL COMMITTEE
ATTN CHIEF, 047
ATTN TECH REPORTS, 013
ATTN PATENT LAW BRANCH, 071
ATTN MCLAUGHLIN, P.W., 741
ATTN LANHAM, C., PROGRAM & PLANS OFFICE
ATTN CHIEF, 0024
ATTN CHIEF, 1020 (20 COPIES)
ATTN CHIEF, 1030
ATTN CHIEF, 1040
ATTN CHIEF, 1050